Listing of Claims

1.-2.   (Canceled)

3.   (Previously Presented) The method of claim 23 wherein the hashing algorithm is a one-way hashing algorithm.

4.   (Previously Presented) The method of claim 23 wherein the first application identifier is a 20-byte hash value.

5.   (Previously Presented) The method of claim 23 wherein the graphical icon data is obtained from an application binary.

6.   (Previously Presented) The method of claim 23 wherein the graphical icon data is obtained from an icon file.

7.   (Previously Presented) The method of claim 23 further comprising comparing the first application identifier with a list of application identifiers to determine an attribute of the software application.

8.   (Original) The method of claim 7 wherein the attribute comprises a parental control rating for the software application.

9.   (Canceled)

10.   (Currently Amended) The method of claim 23 wherein the ~~globally~~ unique identifier indicates that metadata relating to the software application can be obtained from a metadata service.

11.-13. (Canceled)

14.   (Previously Presented) The method of claim 23 wherein the first application

identifier is a unique fixed-length string.

15.   (Currently Amended) The method of claim 23 further comprising storing a relation between the first application identifier and the ~~globally~~ unique identifier in a data file.

16.   (Canceled)

17.   (Previously Presented) The method of claim 23 wherein the distinct application data further comprises registry data.

18.   (Previously Presented) The method of claim 23 wherein the software application is a gaming-related software application.

19.-22. (Canceled)

23.   (Currently Amended) In a computer system, a method of querying a database for information pertaining to a software application installed on the computer system, the software application comprising a plurality of files on the computer system, the files comprising at least an executable file and graphical icon data, the executable file having a name, the method comprising:

in the computer system, generating a first application identifier for a software application, wherein generating the first application identifier comprises generating a hash value with a hashing algorithm applied to distinct application data comprising a combination of at least **a portion of** the graphical icon data of the software application and the name of the executable file for the software application**, wherein the graphical icon data comprises at least one icon that visually represents the software application in a graphical user interface and less than all graphical icon data for the application is used for generating the hash value**;

in the computer system, sending a query for a ~~globally~~ unique identifier for the software application to the database, the query comprising the first application identifier;

in the computer system, receiving a response to the query from the database, wherein the response comprises the ~~globally~~ unique identifier for the software application, and wherein

the ~~globally~~ unique identifier differs from the first identifier; and

in the computer system, displaying information from the response in a graphical user interface.

24.   (Original) The method of claim 23 wherein the database comprises a list of gaming-related software applications, and wherein the response to the query comprises an indicator of whether the software application is a gaming-related software application.

25.   (Previously Presented) The method of claim 23 wherein the query to the database comprises a request for metadata relating to the software application, and wherein the response to the query comprises metadata relating to the software application.

26.   (Previously Presented) The method of claim 23 wherein the query to the database comprises a request for parental control information relating to the software application, and wherein the response to the query comprises parental control information relating to the software application.

27.   (Canceled)

28.   (Currently Amended) In a computer system, a method comprising:

in the computer system, obtaining application data specific to a software application on the computer system, the application data comprising icon data**, wherein the icon data comprises at least one icon that visually represents the software application in a graphical user interface**;

in the computer system, calculating a first value for the application based at least in part on the icon data, wherein calculating the first value comprises applying a hashing algorithm to at least the icon data, and wherein the first value is a hash value;

in the computer system, sending the first value in a database query to a database;

in the computer system, in response to the database query, receiving one or more responses from the database;

in the computer system, obtaining from the one or more responses a ~~globally~~ unique

identifier for the application, wherein the ~~globally~~ unique identifier differs from the first value;

in the computer system, sending the ~~globally~~ unique identifier in a metadata request; and

in the computer system, in response to the metadata request, receiving metadata associated with the application.

29.   (Currently Amended) The method of claim 28 wherein the ~~globally~~ unique identifier is sent in the metadata request ~~sent~~ to a games metadata service, and wherein the metadata is received from the games metadata service.

30.   (Previously Presented) The method of claim 28 wherein obtaining the application data comprises selecting one or more graphical icons from a set of plural graphical icons having different sizes and resolutions.

31.   (Previously Presented) The method of claim 28 wherein the hashing algorithm is a one-way hashing algorithm.

32.   (Previously Presented) The method of claim 28 wherein at least one of the one or more responses indicates that a match for the first value was found in the database.

33.   (Previously Presented) The method of claim 28 wherein the application data further comprises a file name of an application executable, and wherein calculating the first value is further based on the file name of the application executable.

34.   (Currently Amended) The method of claim 28 further comprising:

in the computer system, sending a shortcut link to the database;

wherein the one or more responses comprise a list of possible matches for the shortcut link in the database.

35.   (Currently Amended) The method of claim 28 further comprising:

in the computer system, obtaining from the one or more responses metadata relating to

the software application.

36.   (Currently Amended) The method of claim 28 further comprising:

in the computer system, after receiving the one or more responses, displaying a visual indicator of the software application along with visual indicators of other software applications in a graphical user interface.

37.   (Currently Amended) A computer-readable medium **not consisting of a signal, the computer readable medium** having stored thereon computer-executable instructions operable to cause a computer to perform a method of querying a database for information pertaining to a software application installed on a computer system, the ~~method~~ **computer-readable medium** comprising:

computer-executable instructions for generating a first application identifier for the software application, the software application comprising a plurality of files on the computer system, the files comprising at least an executable file **for the software application** and graphical icon data **for one or more icons associated with the software application**, the executable file having a name, wherein generating the first application identifier comprises generating a hash value with a hashing algorithm applied to **an application-specific binary data block formed from** a subset of the graphical icon data **for one or more icons associated with** ~~of~~ the software application and the name of the executable file for the software application**, and wherein the subset of the graphical icon data comprises at least one icon that visually represents the software application in a graphical user interface**;

computer-executable instructions for sending a query for a ~~globally~~ unique identifier for the software application to the database, the query comprising the first application identifier;

computer-executable instructions for receiving a response to the query from the database, wherein the response comprises an indication of whether the software application is of a particular application type and the ~~globally~~ unique identifier for the software application, and wherein the ~~globally~~ unique identifier differs from the first identifier;

computer-executable instructions for sending a metadata request to a metadata service, the **metadata request** ~~query~~ comprising the ~~globally~~ unique identifier;

computer-executable instructions for receiving metadata for the software application in

response to the metadata request; and

   computer-executable instructions for displaying information from the response to the query from the database along with the metadata received in response to the metadata request in a window of [[a]] **the** graphical user interface along with information for one or more other software applications of the particular application type~~;~~

   ~~wherein the application type is game, and wherein the metadata service is a games metadata service.~~

38.   (New) The computer readable medium of claim 37 wherein the application type is game, and wherein the metadata service is a games metadata service.

39.   (New) The computer readable medium of claim 38 wherein the window of the graphical user interface is part of a gaming activity center, the computer readable medium further comprising:

   computer-executable instructions for determining whether to add the software application to a list of games in the gaming activity center based on the indication of whether the software application is of the particular application type.

40.   (New) The computer readable medium of claim 37 further comprising:

   computer-executable instructions for receiving a second response to the query from the database, wherein the second response comprises an indication that no match for the first application identifier was found in the database; and

   computer-executable instructions for performing a manual search of the database responsive to the indication that no match for the first application identifier was found in the database.